Semi-supervised Approach Based on Co-occurrence Coefficient for Named Entity Recognition on Twitter

Van Cuong Tran, Dosam Hwang
Department of Computer Engineering
Yeungnam University, South Korea
Email: {vancuongqbuni, dosamhwang}@gmail.com

Jason J. Jung*
Department of Computer Engineering
Chung-Ang University, South Korea
Email: j2jung@gmail.com

***Abstract*—The nature characteristics of data in Social Network Services (SNS) are usually short, contain insufficient information, and often are influenced by noise data, thus popular Named Entity Recognition (NER) methods applied for these data could provide wrong results even if they perform well on well-format documents. Most of NER methods are based on supervised learning techniques which often require a large amount of training dataset to train a good classifier. The Conditional Random Fields (CRF) is an example of supervised learning method, which is a statistical modeling method to predict labels for sequences of input samples. Weak point of these method is only perform well on well-format sentences. However the proper sentences are not used frequently in SNS, such as a lot of tweets on Twitter are combinations of independent terms which are implicitly belonged to a context of a certain discussion topic. In this paper, we propose a method to extract named entities from Social Data using a semi-supervised learning method, it is an extension of CRF method which adapts the new challenge with segmentations of data depending on its context rather considering entire dataset. In experiments, The method is applied on a dataset collected from Twitter, which includes 8,624 tweets for training with 1,915 labeled tweets and 1,690 tweets for testing. Our system product a promised result with the F score of the classification result be approximated to 83.9%.**

I. Introduction

Named Entity Recognition (also known as entity identification and entity extraction) is a subtask of information extraction. It seeks entities in documents and classifies them into predefined categories such as person names, locations, organizations, etc [1]. The extracted named entities can be utilized for various purposes such as entity relation extraction, summarizing the documents [2], speech recognition [3], and indexing terms in information retrieval systems [4].

Current NER methods belong to one of the following types: i) Based on predefined rules, which use a named entities list, directories, rule or grammars to identify named entities [5]; ii) Machine learning approaches based on training datasets to automatically extract the named entities [6]; iii) Hybrid methods, which combine above methods and several techniques in natural language processing to achieve better performance [7][8].

Popular NER approaches are either based on linguistic process techniques or statistic methods. Most of them require a large volume of labeled data as training data, and also are designed for a specified language. These approaches achieve good results if they are applied on normal textual documents with proper sentences in terms of grammar and lexicon. However, with the short messages such as tweets, the achievement is not really good with many mistakes. For example, the performance of the Stanford NER that uses the CRF model to train a classifier on CoNLL03 datasets dropped from 90.8% to 45.8% if it is applied on tweets rather than a normal document [9]. The reason of the issue is the characteristics of tweets which are short, informal, ungrammatical, noise and lack of context. The length of tweet is no longer than 140 characters and tweets are containing different kinds of information such as text, links, user mentions (e.g., @BrackObama) and hashtags (e.g., #NewYork). In addition, SNS users often post tweets with free style, acronyms and it is not included extra information to explain detail about the author's opinion. One more challenge related to data in SNS related to its large volume and also its dynamic content in terms of time. Currently, Twitter has more than 288 million monthly active users and 500 million tweets are sent per day[1], the messages on Twitter could be fed into a processing system as a stream of data. It raise a hard challenge to identify entities on the kind of data.

In this paper, we propose a method to recognize named entities on a tweets stream using a semi-supervised approach combined with the CRF model. Firstly, Cosine similarity measurement is applied to cluster unlabeled tweets of training dataset into corresponding groups based on the similarity of its content. Secondly, the CRF model is used to train a classifier on a dataset of labeled tweets. Next, we apply the classifier to label tweets in each cluster. In order to improve performance of the classifier, we propose to consider co-occurrence coefficient of entity candidates and supplementary words, which occur around these candidates. The label of entity is decided depending on the kind of entity has highest score. To deal with the issues of short content and lack of context information, which were not classified entities, the same proper noun in on cluster are considered to label using a same kind label. Finally, the labeled tweets of the clusters are added to the labeled dataset to retrain the classifier.

In order to evaluate our method and show how the system works, we evaluate the model with a training dataset includes 8,624 tweets, in which there are 1,915 tweets are manually labeled. The test dataset contains 1,690 independent tweets. Experimental results shows that our model improves the achievement result.

*Corresponding author

[1] https://about.twitter.com/company

978-1-4673-6640-3/15/$31.00 ©2015 IEEE

Our contributions are summarized as follows:

- We proposed a method that combines the CRF model with a semi-supervised method based on co-occurrence coefficient of supplementary words surrounding proper noun.
- To deal with the lack of context information of data on Twitter, we use Cosine similarity measurement to cluster tweets with similar content into the same group. Using entity's labels which has high confident degree to label similar proper noun, which are low confident.
- We evaluate our method on a human labeled dataset and show that our model outperforms the baseline method by 7.6%.

The outline of this paper is organized as follows. Section 2 presents related works about semi-supervised NER methods. Section 3 describes about our method to solve the problem. Section 4 and 5 show experimental results obtained from our system, and discuss issues for future work.

II. Related works

Named Entity Recognition on Twitter is a hard challenge that attracts more interest from researchers in recent years. The first work that we want to mention here is contributed by Ritter et al. [10]. They rebuild the NLP tool beginning with part-of-speech tagging. The NER method leverages the redundancy inherent in tweets to achieve high performance by using LabeledLDA to exploit Freebase dictionaries as a semi-supervised learning approach. Another recent approach is described by Jung [11], the author proposed three heuristics (i.e., temporal association, social association, semantic association) of contextual association among the microtexts to discover contextual clusters of the microtexts. Instead of examining entire dataset, the NER system is applied for each microtext cluster. As a case study, the author have applied the proposed method on Twitter by using maximum entropy approach-based method which provides 90.3% precision for the best result.

In [9], Liu et al. proposed to combine the K-Nearest Neighbors (KNN) algorithm with the linear CRF model under the semi-supervised approach. The general idea is to use the model to classify tweets in a lexicon level first, and then apply CRFs in order to execute a fine-grained tweet level NER over the results obtained by the KNN algorithm. Finally, 30 gazetteers which cover common names, countries, locations, temporal expression are used to alleviate the lack of the training data.

Named entities tend to occur in the multiple similar tweets, and it is easy to identify them for some of tweets. Liu and Zhou [12] describe two stages labeling system to harvest the redundancy for multiple similar tweets. Firstly, a sequence tagger based on the CRF model labels each tweet. Then it clusters tweets to put the similar tweets into the same groups. Finally, using an enhanced CRF model to refine the labels of each tweet. Li et al. also proposed a novel 2-steps unsupervised NER approach which aim to recognize named entities in Twitter data, called TwinNER [13]. Based on the gregarious property of the named entities in targeted tweets stream. In the first step, it leverages on the global context obtained from Wikipedia and Web N-Gram corpus to partition tweets into valid segments, which are the named entity candidate. In the second step, TwinNER constructs a random walk model to exploit the gregarious property in the local context derived from the Twitter stream. The highly-ranked segments have high opportunity to be true named entities. This approach deals with streams, however it does not determine the class of the identified entity but only identifying if a phrase is an entity or not.

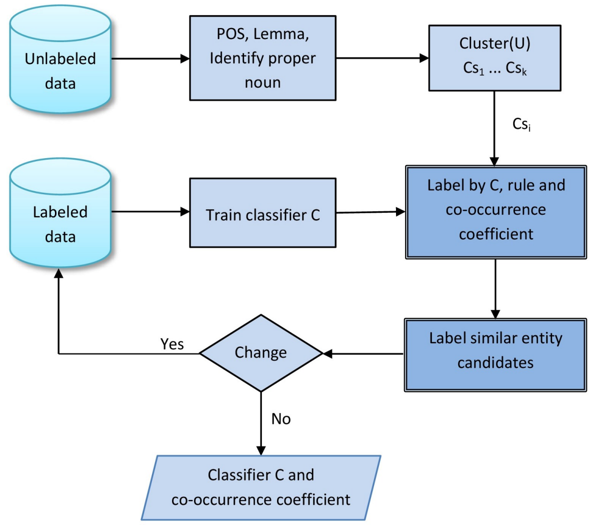
Unlabeled data
POS, Lemma, Identify proper noun
Cluster(U)
$Cs_1 \ldots Cs_k$
Cs_i
Labeled data
Train classifier C
Label by C, rule and co-occurrence coefficient
Label similar entity candidates
Change
Yes
No
Classifier C and co-occurrence coefficient

Fig. 1. The architecture of NER system on Twitter for training phase

In [14], Liao and Veeramachaneni proposed a semi-supervised learning algorithm for NER using the CRF model. The algorithm repetitively learn to improve the training data and the feature set from a small amount of gold data. The trained model is used to extract high confidence data, which is used to discover low-confidence data by using other independent features. These low confidence data then added to the training dataset to retrain the model. They give two ways to obtain independence evidence for entities. The first is based on the fact that multiple mentions of capitalized tokens are likely to have the same label and occur in independently chosen contexts. The second, the entities have context that is highly indicative of the class, however it is independent of the other context (e.g. Inc, Mr., etc). By leveraging such independent evidence, the algorithm can automatically extract high accuracy and non-redundant data for training, and leading to much improved classifier at each iteration.

In this work, our goal is aim to propose a model to extract named entities from tweets on Twitter. It is a semi-supervised learning approach, which combines the CRF model with a classifier based on co-occurrence coefficient of the feature words surrounding the proper noun. We believe that our proposed method is an effective method to solve the problem of NER on Twitter.

III. Basic notions

A. *The tweet*

A tweet is a sequence of tokens defined as follows:

$$tw = (tk_i : i = 1, ..., n) \tag{1}$$

where tk_i is a token in tw; n is the number of tokens of tweet tw.

An example of a tweet: "*What keeps #Chicago small biz owners up at night? They tell @crainschicago: http://t.co/pgpTwYr6bE*", where words begin with "#" character, like *"#Chicago"* is a hashtag, which usually is used to mark keywords or topics in a tweet; the words begin with "@" character, like *"@crainschicago"* is a mention in tweet, that contains another user name; and *"http://t.co/pgpTwYr6bE"* is a shortened link.

B. The named entity candidate

A named entity candidate is a proper noun, which is defined as follows:

$$EC = (tk_i : i = p, ..., q) \tag{2}$$

where:

- $1 \leq p \leq q \leq n$
- tk_p is the word with the POS tagging NNP or the word "The"
- tk_q is the word with the POS tagging NNP
- tk_{p-1}, if exist, does not have POS tagging NNP, the same for tk_{q+1}
- For each i = p, ..., q
  - the POS tagging of tk_i is NNP, or
  - tk_i is an element of set E, where E is the set of special tokens, E = {and, of, for, &, 's}

IV. Proposed method

A. Method Overview

Our proposed method for training and testing phases are described in the Algorithm 1 and 2 respectively. It inherits the idea of Lioa and Veeramachaneni [14] to find new labeled data for the training dataset from unlabeled data based on a semi-supervised learning approach.

Assume that we have a training dataset distributed into two parts: a small set of labeled tweets and a bigger set of unlabeled tweets, which are denoted as L and U respectively. In addition denoting R for all necessary rules to identify entities, which are extracted based on grammatical characteristics of English and writing style of social media messages.

Cosine similarity algorithm [15] is used to cluster tweets into groups of similar tweets in terms of its content. Initially, the labeled training datasets are used to train a classifier based on the CRF model. We used the CRF framework provided by Stanford[2]. This classifier identifies named entities in each cluster. Although the output of this classifier has high precision, the recall is low. To improve the performance, we need to identify additionally named entities which can not be addressed by the classifier.

The architecture of our method is illustrated in the Figure 1. It classifies unlabeled data and then adds them into the labeled training dataset to train the classifier for the next step. The output is continuous to be labeled by using the set of rules R. In order to overcome the limitations of the CRF model, we examine co-occurrences of words which are located around entity candidates. For each proper noun, we measure the average co-occurrence coefficient of pairs between the proper noun and its surrounded words. These value will decide the label of the considering entity candidate.

Dealing with the short length of tweets and the lack of context information, we assume that, tweets in a cluster are related to a same topic which includes mutual entities. Thus, the similar proper noun in different tweets are automatically assigned the same label of detected entities if any. This solution can improve the accuracy of the classifier at the next step iteration. Finally, the labeled tweets of the cluster were added to the labeled training dataset to retrain the classifier.

The number of iterations of the model is depended to the detection result. If there is no more entities to be detected, the training process is finished.

B. Algorithms

The Algorithm 1 describes how the system classifies the unlabeled data. In our work, Part-Of-Speech (POS) Tagger[3] are used to assign the label about the role of a word or a token (e.g. noun, verb, adjective, etc.), and Lemmatizer[4] generate the word lemmas for all tokens in tweets. These packages are publicly available software develop by Stanford. Proper nouns are extracted based on the results of the POS tagging.

Algorithm 1 Algorithm for training phase

```
Input:  $L$ - labeled data
        $U$ - unlabeled data
        $R$ - set of rules
Output: $C$ - classifier
        $\delta$ - co-occurrence coefficient

Tag POS and Lemma $L$, $U$
Get proper noun (PN) and mentions in tweets
$Cs \leftarrow$ Cluster($U$)
repeat
    Use the CRF model to train a classifier $C$ based on $L$
    for each cluster $Cs_i$ in $Cs$ do
        Classify $Cs_i$ by $C$
        Label $Cs_i$ by $R$
        Calculate co-occurrence coefficient $\delta$ of the feature words surrounding named entity
        Label for PN based on the average value of co-occurrence coefficient of the feature words
        Assign similar named entity candidates to have the same label
        Add $Cs_i$ to $L$
    end for
until no new NE can be identified
return $C$, $\delta$
```

[2] http://nlp.stanford.edu/software/CRF-NER.shtml

[3] http://nlp.stanford.edu/software/tagger.shtml

[4] http://nlp.stanford.edu/software/corenlp.shtml

TABLE I. SOME TAG OF POS TAGGER

Tag	Explanation
DT	Determiner
PRP	Personal pronoun
VB	Verb, base form
VBP	Verb, non-3rd person singular present
IN	Preposition or subordinating conjunction
NN	Noun, singular or mass
NNS	Noun, plural
NNP	Proper noun, singular
NNPS	Proper noun, plural

An example of the POS tagging derived from the POS tagger as follows:

Original text: *I'm at Bicycle Ranch in Scottsdale, AZ.*

POS tagging text: *[I/PRP, 'm/VBP, at/IN, Bicycle/NNP, Ranch/NNP, in/IN, Scottsdale/NNP, ,/,, AZ/NNP, ./.]*

In this example, based on the POS tagging of text we can extract three proper nouns are *"Bicycle Ranch"*, *"Scottsdale"*, and *"AZ"*.

For extracting the feature words [16], user mentions are separated, meanwhile hyperlinks, and hashtags are removed from tweets. Stop words are also removed by using the list of predefined stop words[5]. In addition, some kinds of POS tags are not considered such as coordinating conjunction, the cardinal, determiner, symbol.

To present textual information of tweets, a tweet tw is represented by a vector of the feature words with its dimension is equal with the number of the feature words in the training dataset.

$$tw = (w_1, w_2, ..., w_n) \tag{3}$$

where $w_i = 1$ if the word at i^{th} position in the list of the feature words occurs in the tweet tw, otherwise $w_i = 0$; n is the quantity of the feature words in the training dataset;

Textual similarity between two tweets is defined as follows:

$$sim(tw_i, tw_j) = \frac{tw_i \cdot tw_j}{||tw_i||\ ||tw_j||} \tag{4}$$

A tweet tw is assigned into a cluster in case: i) the similarity value between tw and the centroid of cluster is maximum comparing to its distance to other clusters, and ii) the similarity value has to be greater than a defined similarity threshold γ for clustering. Otherwise, a new cluster will be created to contain tw.

In our experiments, the CRF model is used to train a C classifier. At each iteration, the classifier is used to classify the unlabeled data in each cluster. The labeled data of cluster are added to the labeled training dataset to retrain the model.

After labeling cluster Cs_i by classifier C, we continue examining unlabeled proper noun to determine more entity candidates by considering rules in R [17]. The rules have general form as follows:

$$A \mid B \mid C \ \rightarrow D(\sigma) \tag{5}$$

where A is the left-hand side of the considering proper noun (possible empty); B is the considering proper noun; C is the right-hand side of the considering proper noun (possible empty); D is the category of the entity; σ is the confident score of the rule.

For example, if we have a tweet as follows:

"LOVE reading Saunders.10 Stories to Read for Free Online by George Saunders http://t.co/Ku5LfrAz4r via"

and a following rule: If a sentence contains normalized words *"read"*, *"by"* and a proper noun Then the proper noun occurred after the word *"by"* is a person name.

Applying this rule for that tweet, *"George Saunders"* is labeled as "Person".

Algorithm 2 Algorithm for testing phase

```
Input: $U$ - unlabeled data
       $R$ - set of rules
       $C$ - classifier
       $\delta$ - co-occurrence coefficient
Output: $L$ - labeled data
for each tweet $tw \in U$ do
    POS($tw$)
    Get PN, mention in $tw$
    Classify $tw$ by $C$
    Label $tw$ by $R$
    Label for PN based on the average value of co-occurrence coefficient of the feature words
end for
return $L$
```

To overcome the weaknesses of the sequence model, we examine co-occurrence coefficient of the feature words to determine more named entity candidates. With each named entity x_i, we consider the previous feature words and the next feature words of x_i in the sentence $(x_{i-3}, x_{i-2}, x_{i-1}, x_i, x_{i+1}, x_{i+2}, x_{i+3})$. We consider a feature word window, which has length 7, with the named entity is located in the middle of the window. Co-occurrence coefficient of each feature word is calculated by Equation (6):

$$\delta_{x^{\varepsilon}} = \frac{\sum_{i=1}^{k} \frac{1}{\lambda_i}}{n} \tag{6}$$

where ε is the type of entity (Per - Person, Loc - Location, Org - Organization); x is a feature word; λ is the distance from x to named entity; k is the number of sentences which contain x with type ε; n is the quantity of the feature word x in the labeled dataset.

For example, assuming that we have 10 tweets that contain the word *"border"*, in which there are 4 tweets containing named entities tagging with "Location" label as described in Table II. These entities and the word *"boder"* satisfy the condition of the feature word window as above is defined.

Applying Equation (6) to calculate co-occurrence coefficient of the feature word *"boder"* with the type of entity is Location for those four tweets.

$$\delta_{boder^{Location}} = \frac{\frac{1}{2}+\frac{1}{2}+\frac{1}{1}+\frac{1}{1}}{10} = 0.3$$

[5] http://www.textfixer.com/resources/common-english-words.txt

TABLE II. EXAMPLE OF TWEETS AND THEIR FEATURE WORDS

Tweet	Feature words
@FakeSportsCentr yes cuz $Chicago_L$ is a border city #ignorant	yes cuz Chicago be boder city
Man extradited to $U.S._L$ from $Mexico_L$ over slaying of border patrol agent - http://t.co/g544XWzdXM	man extradite U.S. Mexico slay border patrol agent
Leaked photos show immigrant children packed in crowded $Texas_L$ border facilities - http://t.co/t3ybJj7OXY	leak photo show immigrant children pack crowd Texas border facility
This is the worst thing to happen to $Chicago_L$ since border patrol	be worst thing happen Chicago border patrol

We consider the proper nouns, which have not been assigned entity label yet. The average value of co-occurrence coefficient between supplementary words and proper noun are calculated according to each kind of entities as follows:

$$\Psi^{\varepsilon}_{X} = \frac{\sum_{j=1}^{m} \delta_{x_j^{\varepsilon}}}{m} \tag{7}$$

where X is a proper noun; m is the number of the feature words in the considering window.

If Ψ^{ε}_{X} is greater than a threshold α, a suitable label ε will be assigned to X. For example, assuming that we have a tweet *"Do you have to cross a border to go to Scotland? - Spencer the droman"*. In this tweet, the word *"Scotland"* is a proper noun and considered to be a named entity candidate. We apply Equation (7) to identify its type by calculating the average of co-occurrence coefficient of its neighbors.

$$\Psi^{Per}_{Scotland} = \frac{\delta_{cross^{Per}} + \delta_{border^{Per}} + \delta_{go^{Per}}}{3}$$

$$\Psi^{Loc}_{Scotland} = \frac{\delta_{cross^{Loc}} + \delta_{border^{Loc}} + \delta_{go^{Loc}}}{3}$$

$$\Psi^{Org}_{Scotland} = \frac{\delta_{cross^{Org}} + \delta_{border^{Org}} + \delta_{go^{Org}}}{3}$$

Assume that the highest value of three above values is $\Psi^{Loc}_{Scotland}$ and it is greater than the threshold α, then the label "Location" is assigned to "*Scotland*".

In addition, in SNS an account name of user has high probability to include their real name or casual name. In Twitter by mentioning an username in tweets, author links their part of content to his/her friend's account using a simple syntax such as *@BarackObama*, *@TheDemocrats*, *@Chicago_Reader*, etc. So that, if a proper noun in a tweet is occurred in the mention content, it is potential be a named entity. Thus in this case, its score according to the corresponding category should be increased a certain β times. For example, we have a tweet:

"*@ArianaGrande Hey Ari Would you mind retweeting the Tweet I Retweeted to Get @justinbieber To Follow #oomf*"

Supposing that we set threshold $\alpha = 0.3$, $\beta = 3$ and we calculate the average value of co-occurrence coefficient of feature words surrounding the proper noun "Ari" as follows:

$\Psi^{Per}_{Ari} = 0.114$; $\Psi^{Loc}_{Ari} = 0.034$; $\Psi^{Org}_{Ari} = 0.002$

All of three above value is not satisfy the threshold α for a type of entity but "Ari" contain in mention $@ArianaGrande$ of this tweet so the average value of co-occurrence coefficient is multiplied 3. In that case, Ψ^{Per}_{Ari} is highest value due to the final score is become $0.114 * 3 = 0.342$. This value is greater than threshold α, thus the label "Person" is assigned to "Ari".

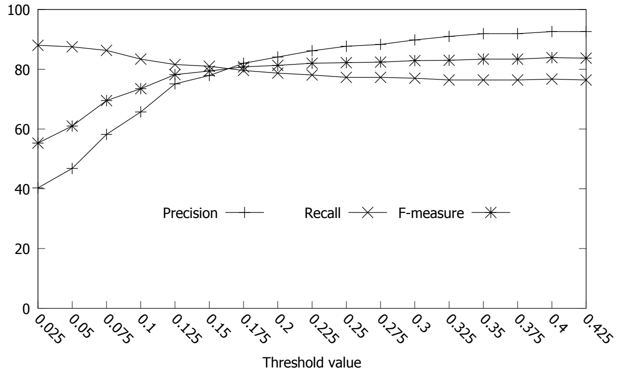

Fig. 2. Performance of system with respect to co-occurrence coefficient

Tweets in one cluster are processed to assign labels in each iteration. These tweets are similar context. Some of these tweets are too short and lack of context so they are hard to classify named entity by the CRF model or co-occurrence coefficient measurement of surrounding feature words. We force the label of the same proper noun in one cluster to become the same label. This solution is to avoid inconsistencies in the labeled training dataset.

V. EXPERIMENTAL RESULTS

In order to evaluate the performance of our proposed method, we use the Java library for the Twitter API[6] to collect tweets of 13 users from January 1^{th} 2013 to December 10^{th} 2014, then drop tweets which are only hashtags, mentions, hyperlinks or emoticon. Finally, 10,314 tweets are selected for our work. The data is distributed into two separate sets for training and testing phase. The training dataset includes 8,624 tweets collected from January 1^{th} 2013 to October 31^{th} 2014, in which 1,915 tweets are labeled. The testing dataset includes 1,690 tweets collected from November 1^{th} 2014 to December 10^{th} 2014.

We manually annotate the given dataset with three types of named entities: "Organization", "Person", and "Location". A non-named entities are annotated as "Other", which means that is other type. Each word token in labeled data is marked with </Type>, where "Type" is the type of the entity[7]. In our labeled dataset, 1,915 tweets are manually annotated for training phase and 1,690 tweets are annotated to form the gold-standard dataset for evaluation. Performance on this task is evaluated by measuring the Precision (8) and the Recall (9) of labeled entities, combined into the F-measure (10).

$$Precision = \frac{True\ positive}{True\ positive + False\ positive} \tag{8}$$

$$Recall = \frac{True\ positive}{True\ positive + False\ negative} \tag{9}$$

$$F - measure = 2\ .\ \frac{Precision\ .\ Recall}{Precision + Recall} \tag{10}$$

[6] http://twitter4j.org

[7] http://nlp.stanford.edu/software/crf-faq.shtml

TABLE III. EXPERIMENTAL RESULTS

	Precision(%)	Recall(%)	F-measure(%)
Baseline	92.9	64.7	76.3
Our system	92.6	76.7	83.9

In this paper, the output results by applying CRF model is considered as a baseline for comparison. The results of our method and the baseline are compared in Table III. For the achievement, we set some values of β, γ, and the suitable value is $\beta = 3$ and $\gamma = 0.05$.

In order to find out the impact of co-occurrence coefficient to system's performance, we evaluate the system with several the α threshold values. We calculate the performance of the NER task as shown in Figure 2. The precision is improved when threshold α increases, and the recall also decreases but not more. The overall, F-measure is improved when we increase the value of the threshold α.

As shown in Table III, our system obtained the promised performance. It reaches highest F-measure (83.9%) with the value of the threshold $\alpha = 0.4$. F-measure score is higher than the baseline 7.6%, meanwhile the recall is higher than the baseline 12%. It means that our model improves the performance of the CRF model even with limited training dataset.

VI. CONCLUSION

In this paper, we presented the semi-supervised learning approach for extracting named entities from tweets. The method is a combination of several techniques such as sequential classification based on CRF, semantic rules, co-occurrence coefficient of feature words and proper noun. The experimental results show that our proposed method achieves high performance with only a small amount of labeled training dataset and improve the recall of the sequence label model CRF.

In the future, we will improve our proposed method by using extra knowledge base such as Gazetteers, Freebase and so on. In addition, improving the training process by collect related information for understanding context of the given data such as the data in websites embedded in tweets as hyperlinks.

ACKNOWLEDGMENT

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (NRF-2014R1A2A2A05007154). Also, this work was supported by the BK21+ program of the National Research Foundation (NRF) of Korea.

REFERENCES

[1] S. Abdallah, K. Shaalan, and M. Shoaib, "Integrating rule-based system with classification for arabic named entity recognition," in *Computational Linguistics and Intelligent Text Processing*. Springer, 2012, pp. 311–322.

[2] C. Nobata, S. Sekine, H. Isahara, and R. Grishman, "Summarization system integrated with named entity tagging and ie pattern discovery." in *Proceedings of Third International Conference on Language Resources and Evaluation*, 2002, pp. 1742–1745.

[3] C. Meyer and H. Schramm, "Boosting hmm acoustic models in large vocabulary speech recognition," *Speech Communication*, vol. 48, no. 5, pp. 532–548, 2006.

[4] H.-H. Chen, Y.-W. Ding, and S.-C. Tsai, "Named entity extraction for information retrieval," *Computer Processing of Oriental Languages*, vol. 12, no. 1, pp. 75–85, 1998.

[5] L. Chiticariu, R. Krishnamurthy, Y. Li, F. Reiss, and S. Vaithyanathan, "Domain adaptation of rule-based annotators for named-entity recognition tasks," in *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2010, pp. 1002–1012.

[6] J. R. Finkel and C. D. Manning, "Nested named entity recognition," in *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*. Association for Computational Linguistics, 2009, pp. 141–150.

[7] M. Van Erp, G. Rizzo, and R. Troncy, "Learning with the web: Spotting named entities on the intersection of nerd and machine learning." in *# MSM*. Citeseer, 2013, pp. 27–30.

[8] D. Küçük and A. Yazıcı, "A hybrid named entity recognizer for turkish," *Expert Systems with Applications*, vol. 39, no. 3, pp. 2733–2742, 2012.

[9] X. Liu, S. Zhang, F. Wei, and M. Zhou, "Recognizing named entities in tweets," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, 2011, pp. 359–367.

[10] A. Ritter, S. Clark, O. Etzioni *et al.*, "Named entity recognition in tweets: an experimental study," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2011, pp. 1524–1534.

[11] J. J. Jung, "Online named entity recognition method for microtexts in social networking services: A case study of twitter," *Expert Systems with Applications*, vol. 39, no. 9, pp. 8066–8070, 2012.

[12] X. Liu and M. Zhou, "Two-stage ner for tweets with clustering," *Information Processing & Management*, vol. 49, no. 1, pp. 264–273, 2013.

[13] C. Li, J. Weng, Q. He, Y. Yao, A. Datta, A. Sun, and B.-S. Lee, "Twiner: named entity recognition in targeted twitter stream," in *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2012, pp. 721–730.

[14] W. Liao and S. Veeramachaneni, "A simple semi-supervised algorithm for named entity recognition," in *Proceedings of the NAACL HLT 2009 Workshop on Semi-Supervised Learning for Natural Language Processing*. Association for Computational Linguistics, 2009, pp. 58–65.

[15] J. Yin, "Clustering microtext streams for event identification," in *Proceedings of the Sixth International Joint Conference on Natural Language Processing*. Asian Federation of Natural Language Processing, 2013, pp. 719–725.

[16] H. L. Nguyen, T. D. Nguyen, D. Hwang, and J. J. Jung, "Kelabteam: A statistical approach on figurative language sentiment analysis in twitter," in *Proceedings of the 9th International Workshop on Semantic Evaluation*. Association for Computational Linguistics, 2015, pp. 679–683.

[17] R. Alfred, L. C. Leong, C. K. On, and P. Anthony, "Malay named entity recognition based on rule-based approach," *Int. J. Mach. Learn. Comput*, vol. 4, no. 3, pp. 300–306, 2014.